Energy Harvest (E-harv) Model Introduction

- **E-harv Model is a built-in library in** SUPR Model[1] **in Matlab-Simulink**

- **Goal of E-harv Model is to accurately model functionality, energy, and efficiency of functional components of energy scavenging systems. We plan to use it to simulate energy supply to signal paths on chip through dynamic power management (DPM), and then intelligently make architectural design decisions**
- **The model is still in the process of being improved and completed**

Harvesting System Components

1. Below is a block diagram showing the energy flow in the system. From left to right, we have input from environmental dynamics (e.g. thermal differences or solar power), harvesting device (e.g. thermoelectric generator (TEG)), energy storage (e.g. capacitor or rechargeable battery), voltage regulators, and then system on chip.

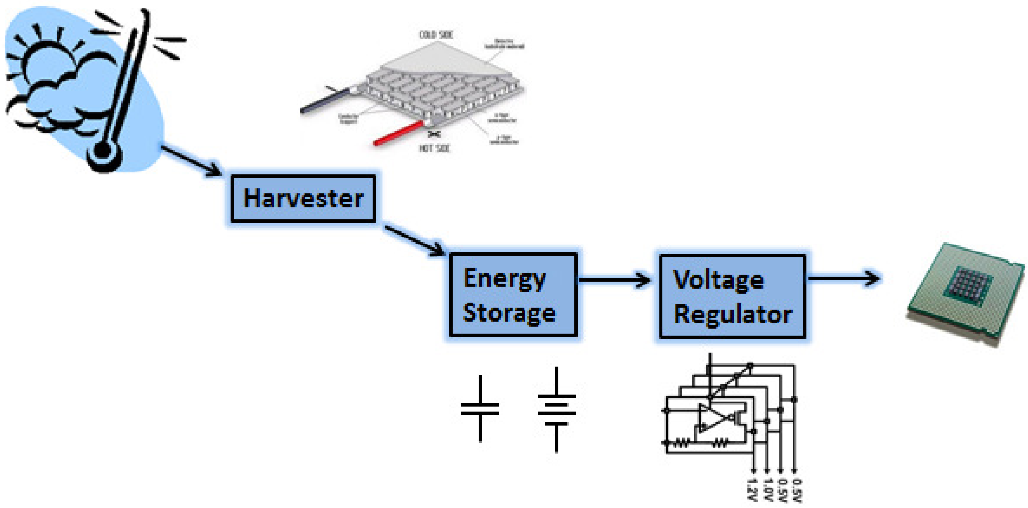


Figure.1 Energy Scavenging System Block Diagram

2. A closer view of the energy storage block: we combined the boost converter function block with the storage component. **Figure 2** is the block diagram of the *BC+Capacitor* E-harv block, which will be used for testing illustrations later, with key parameters labeled.

[1] Information about SUPR Model – please go to RLP-VLSI group Wiki SUPR Model Tutorials https://venividiwiki.ee.virginia.edu/mediawiki/index.php/SUPR_Model

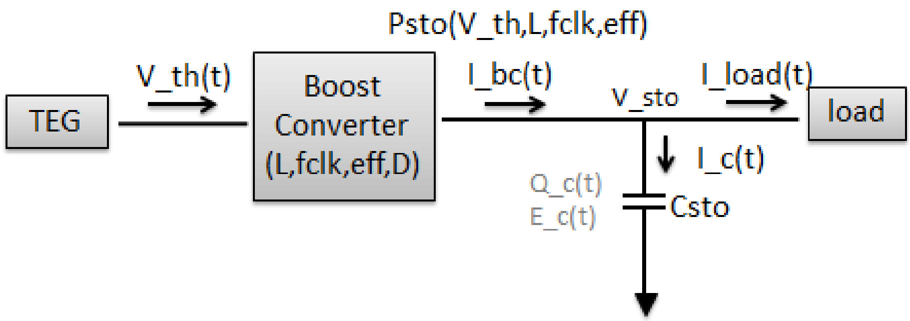


Figure.2 E-harv: *BC+Capacitor* Block Diagram with Key Parameters

3. Model is completely in Simulink. We've created a library of blocks that can be used for energy harvesting related testing. It is called "Energy Harvest" within SUPR:

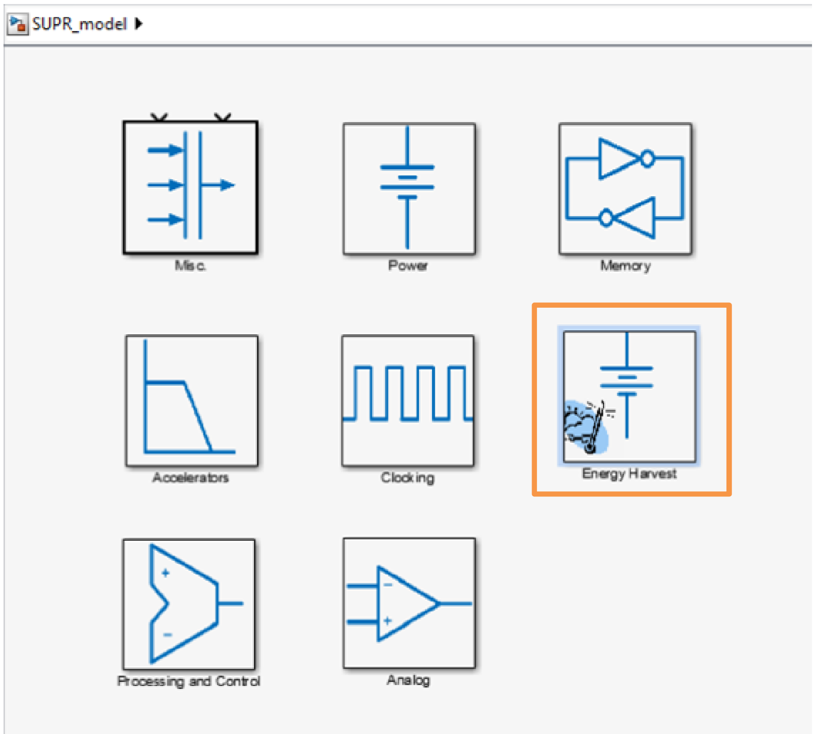


Figure.3 Blocks in SUPR Model

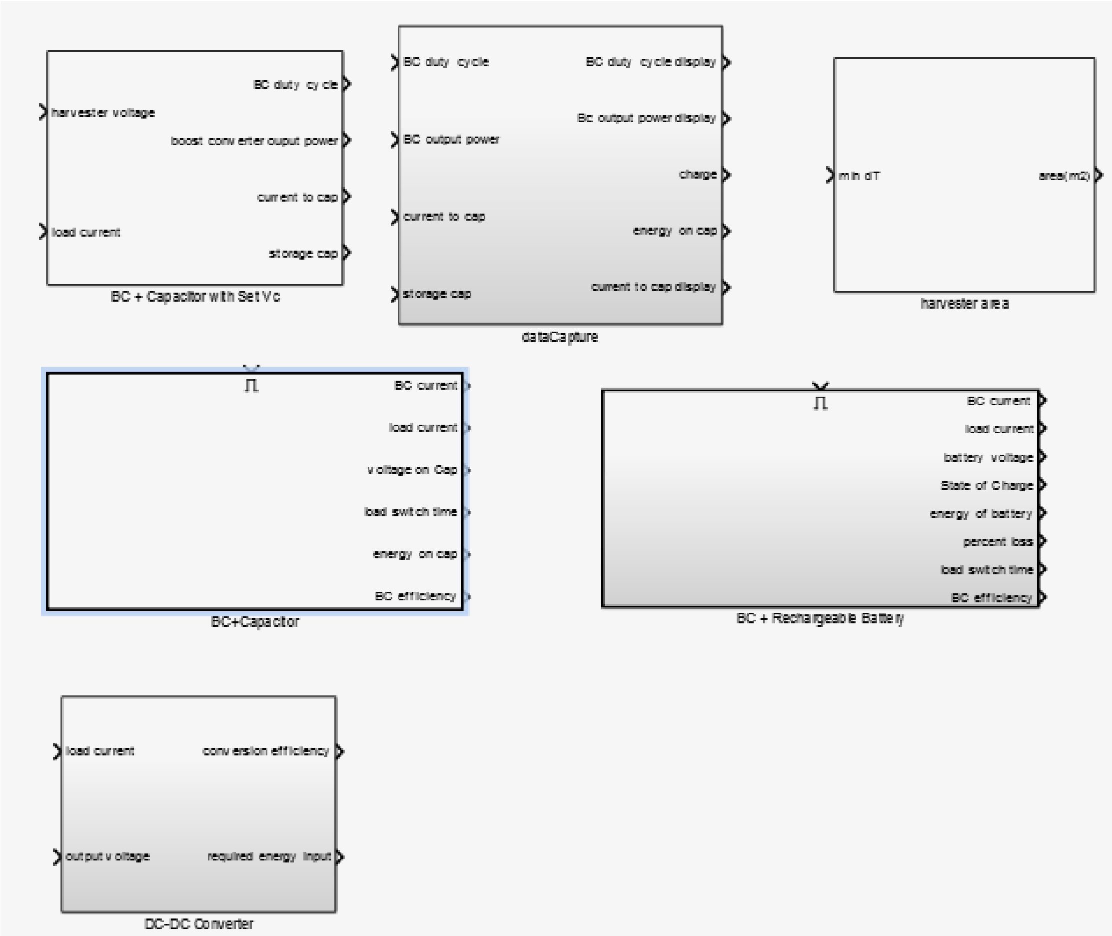


Figure.4 Blocks in Energy Harvest Library

The following blocks exist under Energy Harvest:

i. **BC + Capacitor with Set Vc**: boost converter and storage capacitor; assuming storage capacitor voltage is at a pre-defined target value (vc)
ii. **harvester area** : harvesting device minimum area calculations
iii. **dataCapture**: hides data capture connections for **BC + Capacitor with Set Vc**
iv. **BC + Capacitor**: boost converter and storage capacitor; general
v. **BC + Rechargeable Battery**: boost converter and rechargeable battery as storage
vi. **DC-DC Converter**: efficiency calculations

Using the Model: How to Run a Test

Here, we documented the procedures of running a test using the E-harv model.

Test:

- Vary TEG harvested voltage over time (hardware)
- Vary load current over time (power being drawn by the load)
- Plot different output voltage vs time profiles

E-harv block used: **BC + Capacitor**

1. Have Matlab(Simulink) installed on your computer - http://its.virginia.edu/research/matlab/

2. Find folder *SUPR_model* under Dropbox – Bengroup or Linux server – Bengroup using the following directories:

Dropbox	User>Dropbox>Bengroup>SUPR E-Harv>SUPR_model
Linux	Filesystem>homenfs>platou>bengroup>workspace>Chu>SUPR_model

Copy *SUPR_model* and put it into you Matlab folder: Libraries>Documents>MATLAB>SUPR_model; open Matlab

3. After opening Matlab, in the Current Folder window below, right click on the *SUPR_model* folder *>Add to Path> Selected Folder and Subfolders*

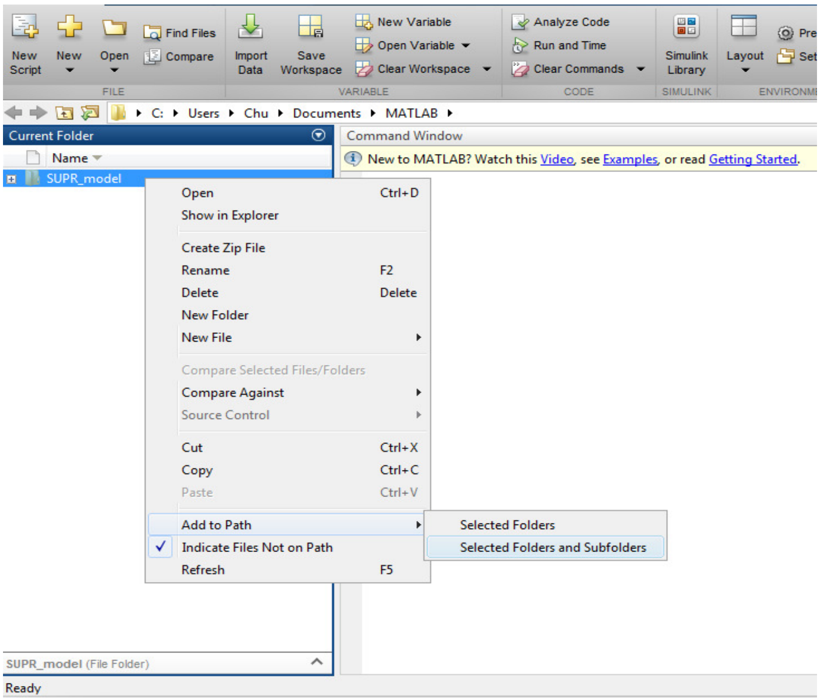
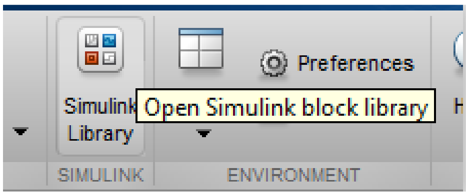


Figure.5 Adding Folder to Path

4. Open Simulink:



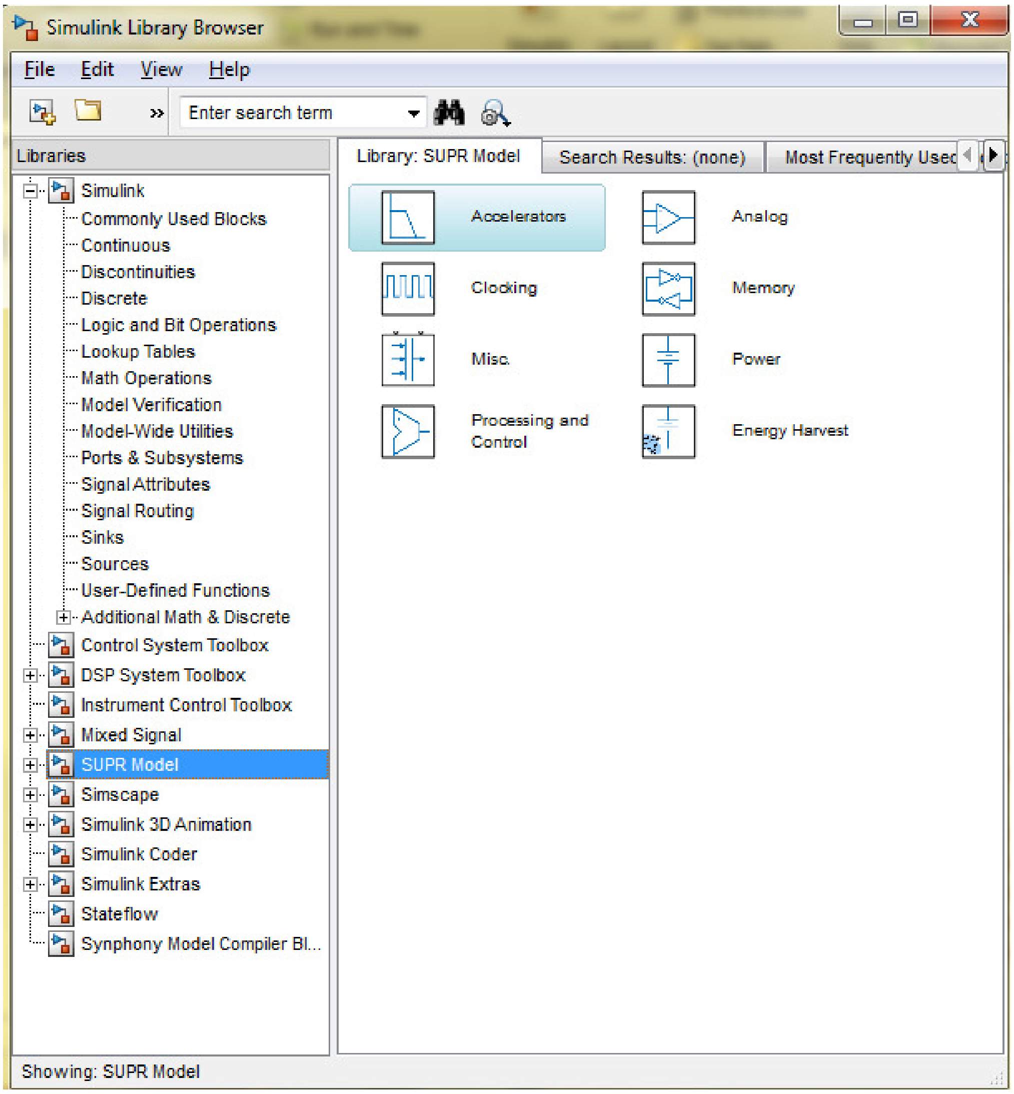


Figure.6 Simulink Library Browser

In the Simulink Library Browser (**Figure 6**), find **SUPR Model**. Right click on it and choose *Open SUPR Model Library*. New window should pop out.

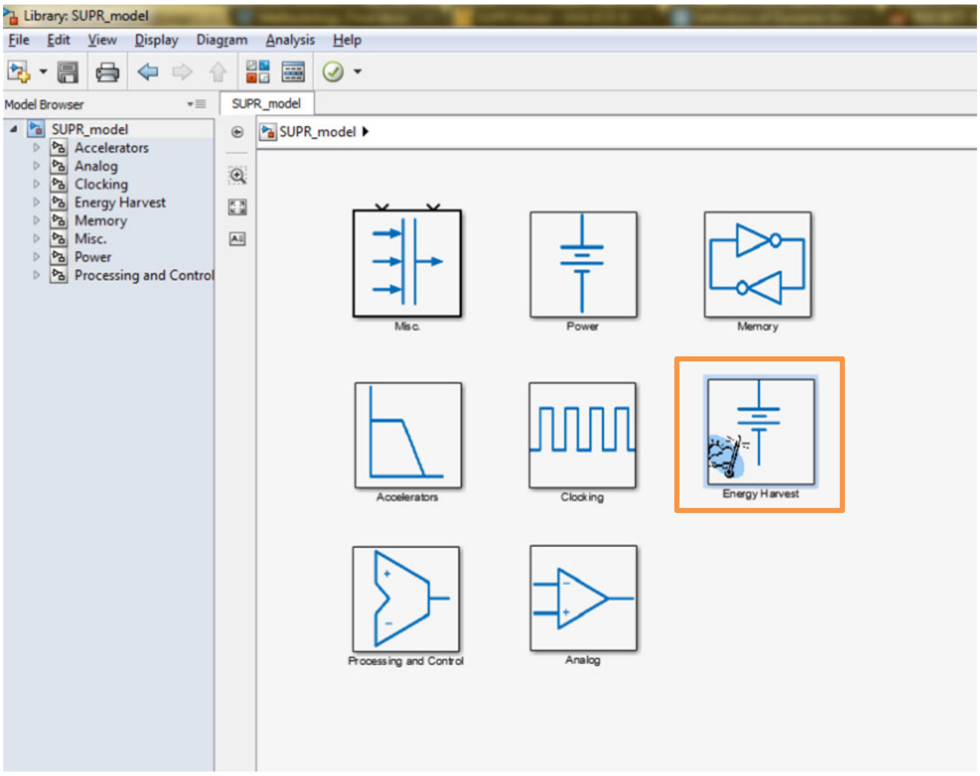


Figure.7 SUPR Model Blocks

Double click on the **Energy Harvest** block (**Figure 7**) to enter the E-harv library (**Figure 8**):

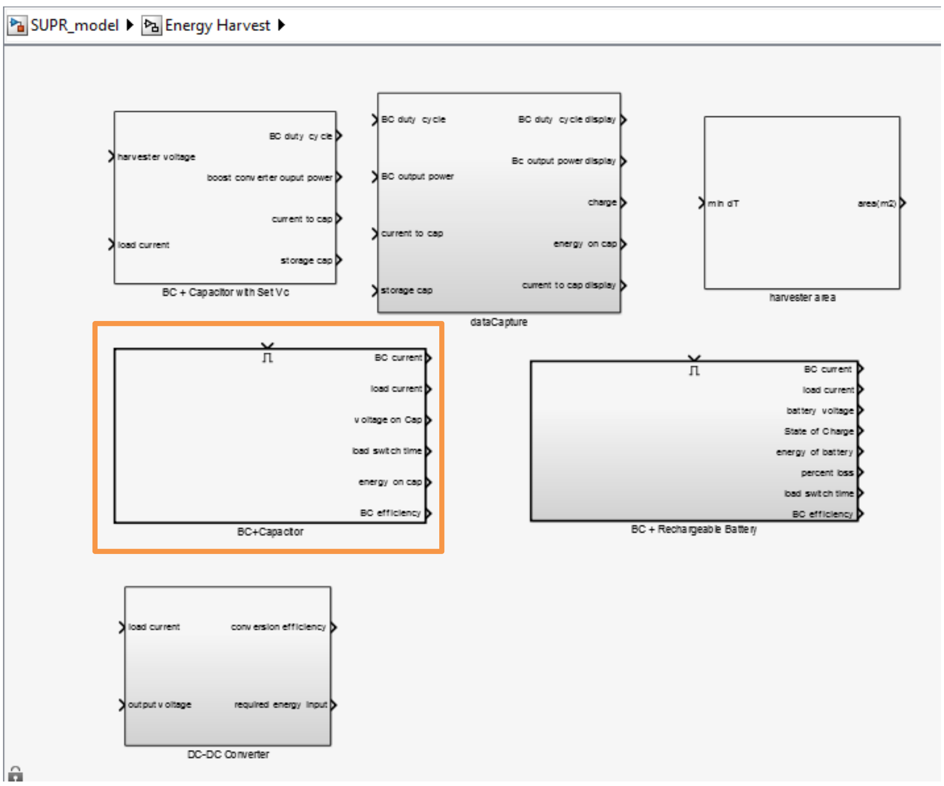
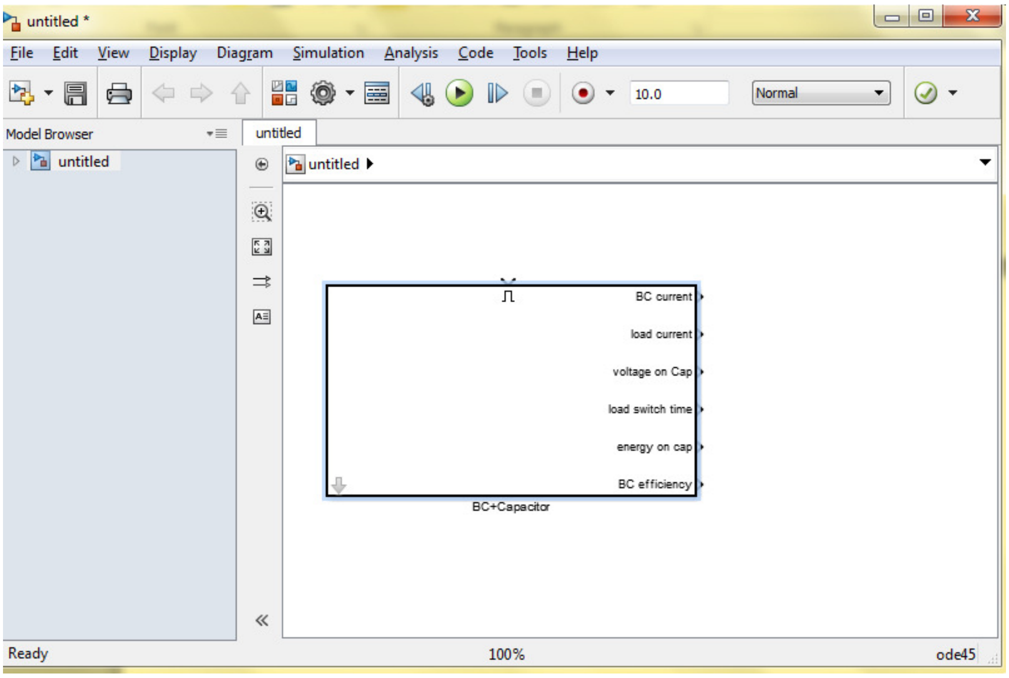


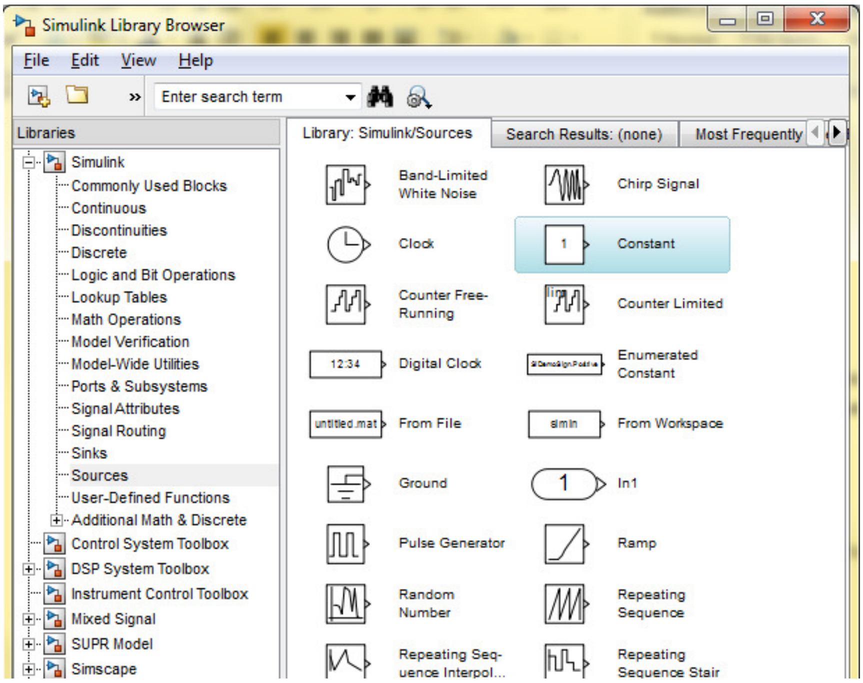
Figure.8 Energy Harvest Library

Set up Test

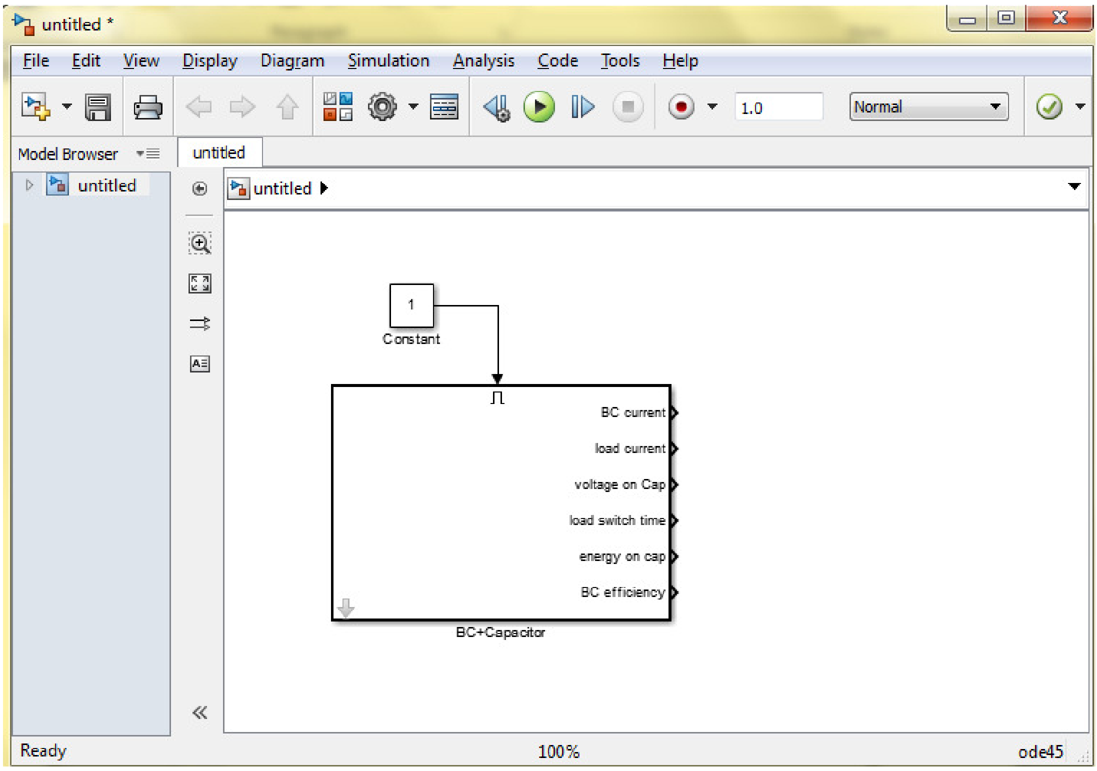
5. Go to *File>New>Model* to create a new simulation space. Click-and-drag **BC-Capacitor** Block from the library to the new model *untitled**(later remember to rename and save)



The only needed input for this block is an enable as shown by the clock symbol at the top. Go to *Simulink library>Sources>Constant:*



Put it in the *untitled** new model, and connect it to the enable input of **BC + Capacitor.** Now we have:



Data Input from CSV File

6. For this test, we assume we have valid TEG output voltage, current, and SoC load current profiles over time as streams of input data to the block. The hidden script (S_function) of **BC + Capacitor** can read arrays of voltage & current values from a CSV file named *harvester_data*[2] in the columns format:

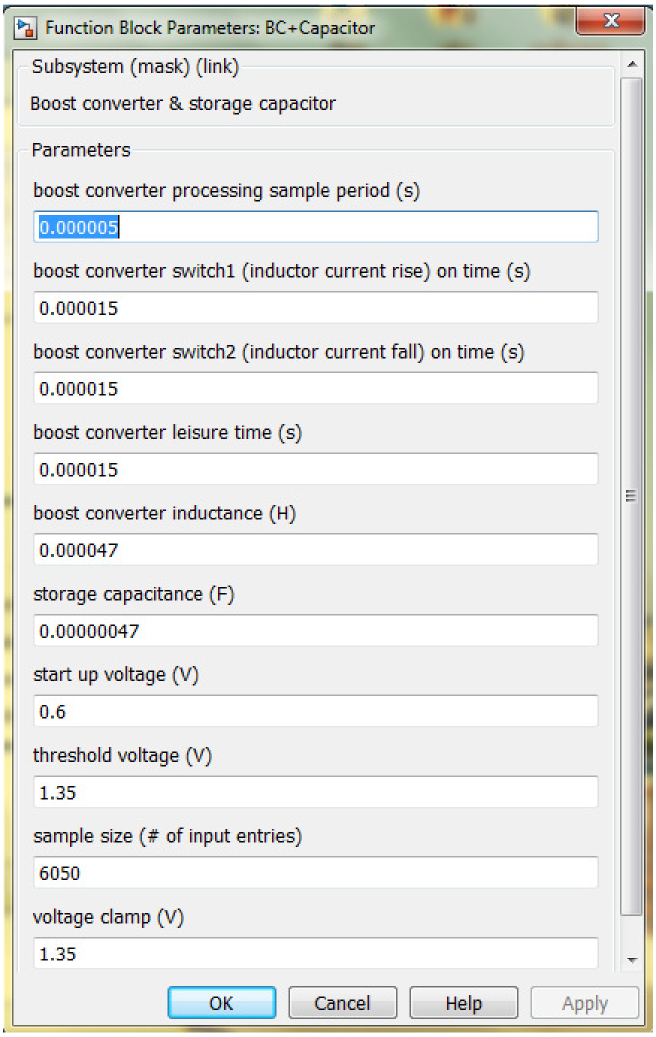
Time value 0	Voltage value 0	TEG current value 0	Load current value 0
Time value 1	Voltage value 1	TEG current value 1	Load current value 1
Time value 2	Voltage value 2	TEG current value 2	Load current value 2
…	…	…	…
..	..	..	..

[2] This CSV file must be stored in the directory: MATLAB>SUPR_model>CSV>*harvester_data*

Please make sure these 4 columns of data have the **same size**.

Parameter

7. Now we have the test environment and the inputs ready. Go to *untitled**model, double click on **BC + Capacitor**. You will see the following parameter input window:



These parameters are very essential to the system. How we vary the values of them can change the output performance greatly. Here we are using the following parameters:

Simulation Parameters and Conditions: (Can be changed from user-input interface)[3]

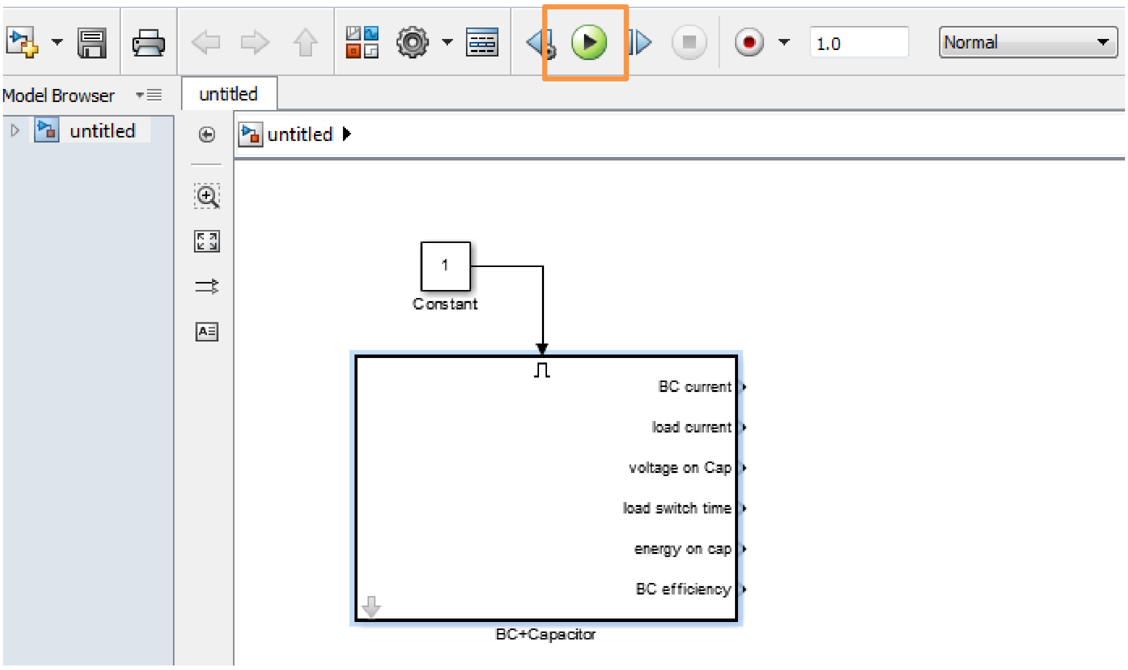
---

[3] Relevant parameters and their meaning can be found in research paper mentioned at the end of this documentation, or through other relevant resources

- Processing frequency: 200kHZ (Tsample = 5 us)
- Number of samples: 6050 (0.035s)
- Boost converter switch 1 on time = switch 2 on time = idle time = 3*Tsample
- Boost converter inductance: 47 uH
- Storage capacitor : 47nF
- Assuming TEG voltage is constant (0.005 V over time)
- Node threshold voltage = clamp voltage = 1.35V
- Start-up voltage = 600mV

Running the test

8. After successfully setting up the environment, input data, and parameters, we are ready to run the test. Click on the green "play" button in the tool bar:

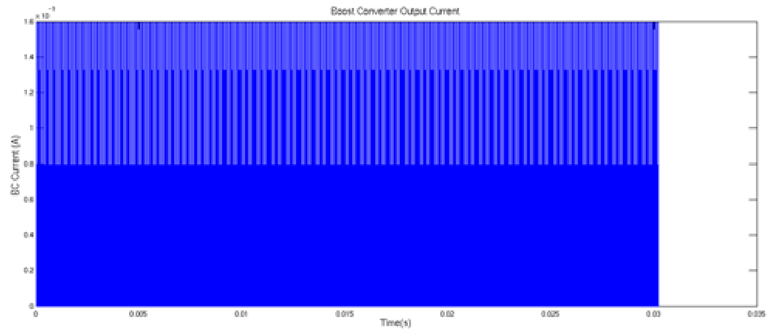


Results

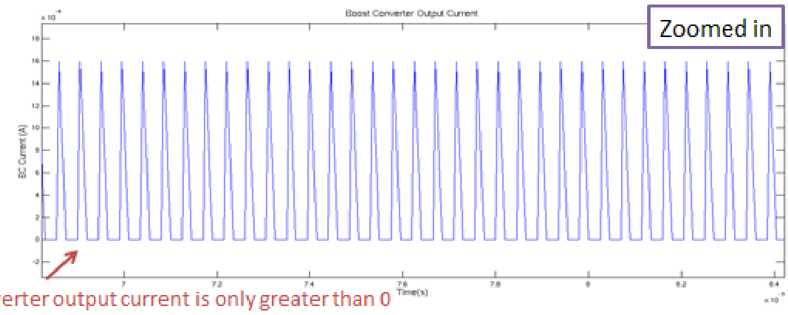
Plots of different outputs will pop out within 1 second (for 6050 samples).

Among them, we can see:

Boost Converter Output Current



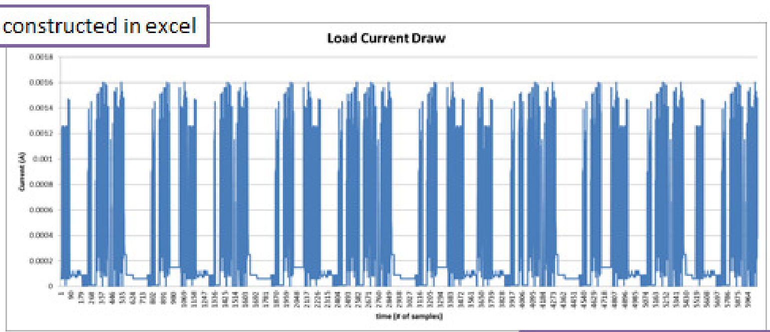
Zoomed in



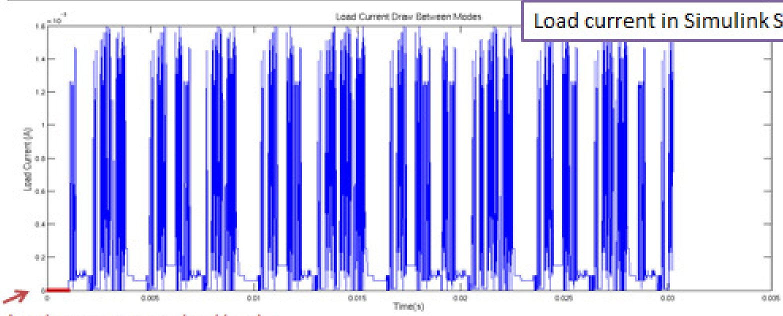
Boost converter output current is only greater than 0
when switch 2 is on & switch 1 is off

Load Current Draw from Storage Capacitor

Load current constructed in excel

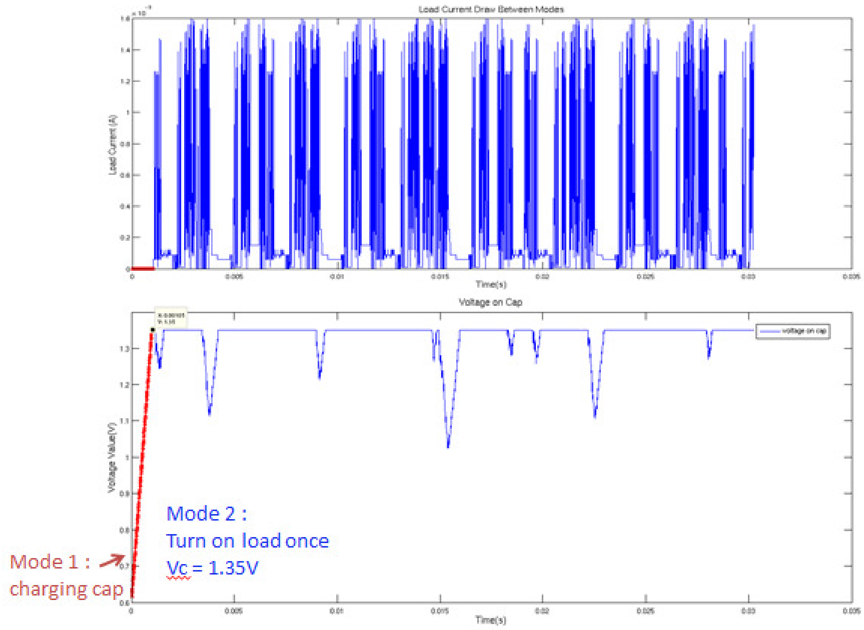


Load current in Simulink Simulation



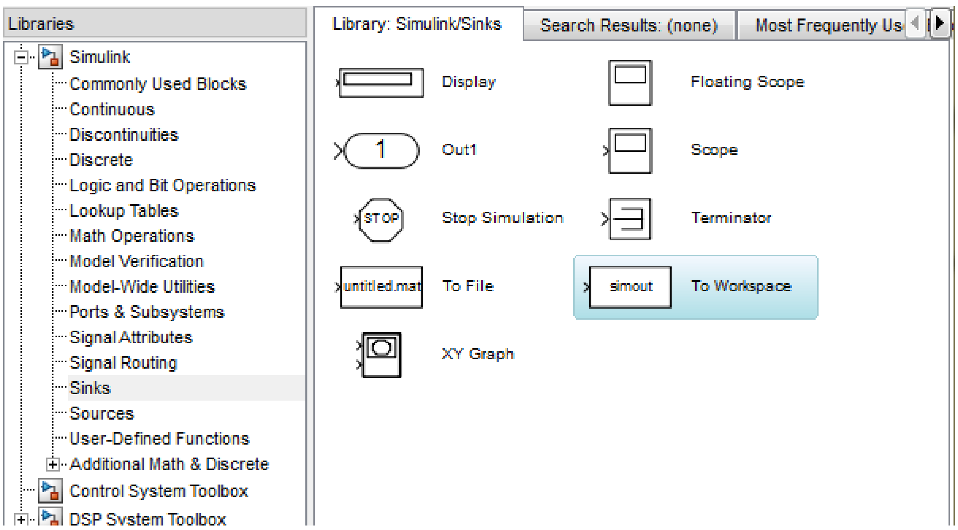
In simulation, load current is pushed back:
load is turned on when Vcap > = Vthreshold

Adjusted Load Current and Corresponding Node Voltage

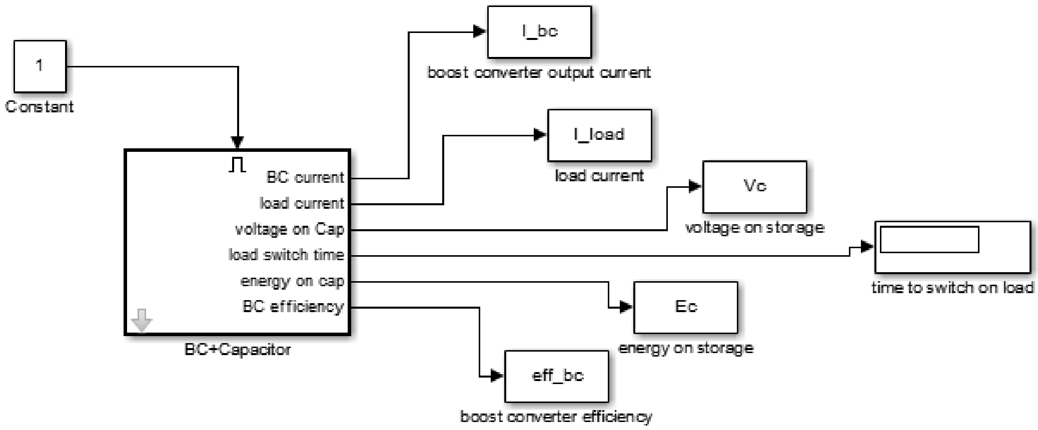


We can also output results and store them in the form of arrays through outputting to Matlab Workspace or Excel files. Below we give the instructions for outputting data to workspace.

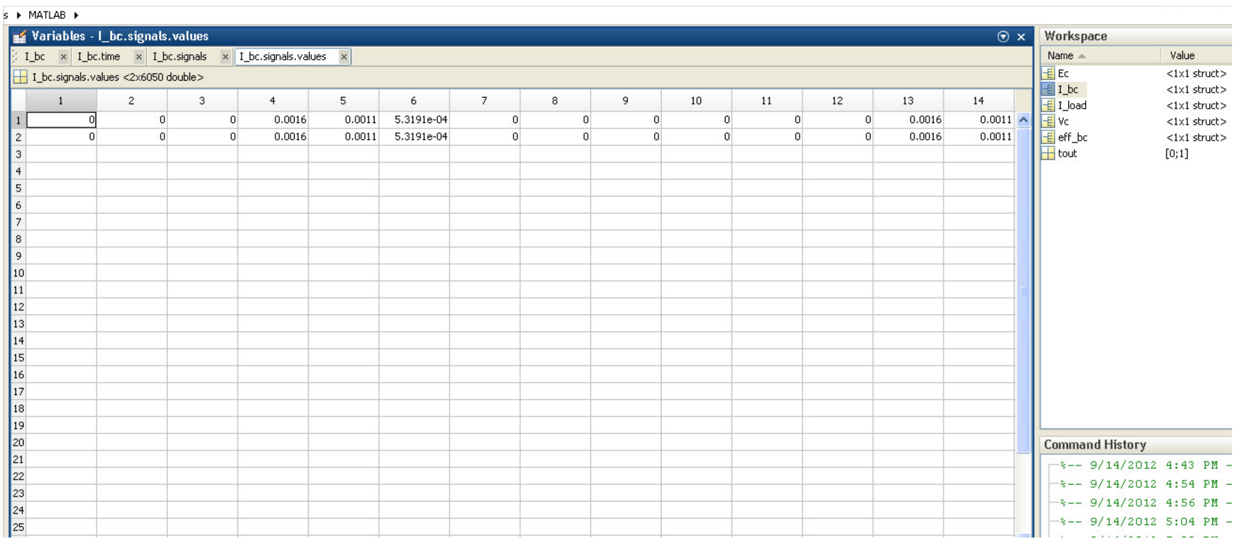
From *Simulink library>Sinks>To Workspace,* add output-# (6 here) dataCapture *To Workspace* to the *untitled** model, and connect each BC-Capacitor block output port to each *To Workspace*



Now we have:



With the correct inputs & parameters, click *Run*. After the simulation finishes, go to Matlab command window.



In the Workspace window at the right top corner, we can see stored output variables. Click on any of them, we can see their time step to time step value overtime (left window).

The simulated results can also be stored to a file or taken as input to another block. More information on Simulink data capture - http://www.mathworks.com/academia/student_center/tutorials/

Model Capabilities & Future Plans

Good Model Paper References